

Cvičení #8: (m, c) -stromy

Miloš Chromý

chromy@ktiml.mff.cuni.cz

1. **Do duše oknomedián.** Máme nekonečnou posloupnost, číslo k a dvě operace, které nad posloupností chceme provést. Operace **načti**(x) načte další znak z posloupnosti. Operace **k -medián**() vypíše medián z posledních k prvků. Navrhněte algoritmus pro každou z operací s co nejlepší časovou složitostí. (**načti**(x) v $O(\log k)$ a **k -medián**() v $O(\log k)$).
2. **Průmět.** Modifikujte AVL strom tak, aby uměl odpovídat na dotaz průměr všech prvků z intervalu (a, b) s logaritmickou časovou složitostí. Složitost všech ostatních operací musí zůstat stejná.
3. **(a, b) -stromy.** (a, b) -stromy se používají často pro práci s většími daty. Cíl je, aby se do paměti vešel konstantní počet vrcholů (a, b) -stromu. Načtení nového vrcholu, který není v paměti je mnohem dražší operace než libovolná operace na vrcholech, které jsou již uloženy v paměti. Z přednášky znáte operace **insert** a **delete**, které štěpí/slučují vrcholy, pokud v průběhu operací narušíte strukturu (a, b) -stromů. Bohužel tyto operace se nejdříve zanoří do listů a pak se vrací do kořene a načítají tedy každý vrchol při průchodu dolů i nahoru. Navrhněte implementaci takovou, která při operaci **insert** a **delete** projde strom jen jednou od kořene do listu.
4. **Join.** Navrhněte operaci **join**(X, Y), která dostane dva (a, b) -stromy X a Y a sloučí je do jednoho. Může se přitom spolehnout na to, že všechny klíče z X jsou menší než všechny z Y . Zkuste dosáhnout složitosti $O(\log |X| + \log |Y|)$.
5. **Split.** Navrhněte operaci **split**(T, x), která zadaný (a, b) -strom T rozdělí na dva stromy. V jednom budou klíče menší než x , v druhém ty větší. Pokuste se o logaritmickou časovou složitost.
6. **Prázdná paměť.** Nevýhodou (a, b) -stromů je, že plýtvají pamětí – může se stát, že vrcholy jsou zaplněné jen z poloviny. Navrhněte úpravu, která zaručí zaplnění z alespoň $2/3$.
7. **Minimový strom.** Minimový binární strom $M(a)$ pro zadanou posloupnost $a = (a_1 \dots a_n)$ bez opakujících se prvků je definován takto: Buď a_i minimální prvek $a = (a_1 \dots a_n)$; pak kořen $M(a)$ je a_i , jeho levý podstrom je $M(a_1 \dots a_{i-1})$ a pravý podstrom $M(a_{i+1} \dots a_n)$. $M(\emptyset)$ je prázdný strom. Pro zadanou posloupnost postavte minimový strom v čase $O(n)$.